LEVEL II

12 5c

INTERACTIVE AIDS FOR CARTOGRAPHY AND PHOTO INTERPRETATION

Semiannual Technical Report
Covering the Period May 11, 1978 to October 11, 1978

SRI Project 5300
Contract DAAG29-76-C-0057

ARPA Order No. 2894-5
Program Code No. 61101E

Contract Amount: $1,373,527
Effective Date: May 12, 1976
Expiration Date: October 9, 1979

By: Martin A. Fischler, Senior Computer Scientist
Principal Investigator
(415) 326-6200, Ext. 5106

and

G. J. Agin
H. G. Barrow
R. C. Bolles
M. A. Fischler
L. Quam
J. M. Tenenbaum
H. C. Wolf

Prepared for:

Director
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209

Attention: Program Management

SRI International
333 Ravenswood Avenue
Menlo Park, California 94025
(415) 326-6200
Cable: SRI INTL MNP
TWX: 910-373-1246

SRI International

INTERACTIVE AIDS FOR CARTOGRAPHY AND PHOTO INTERPRETATION.

(9) Semiannual Technical Report. 11 May-11 Oct 78,
Covering the Period May 11, 1978 to October 11, 1978

SRI Project 5300
Contract DAAG29-76-C-0057 (15) ARPA Order-2894

ARPA Order No. 2894-5
Program Code No. 61101E

Contract Amount: $1,373,527
Effective Date: May 12, 1976
Expiration Date: October 9, 1979

(11) 1978

(10) By: Martin A. Fischler, ~~Senior Computer Scientist~~
~~Principal Investigator~~
~~(415) 326-6200, Ext. 5106~~

L. Quam,
G. J. Agin,
H. G. Barrow
R. C. Bolles
~~M. A. Fischler~~

and

L. Quam
J. M. Tenenbaum
H. C. Wolf

(12) 63 p.

Prepared for:

Director
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209

Attention: Program Management

410 281

Approved:

Peter E. Hart, Director
Artificial Intelligence Center

David H. Brandin, Executive Director
Computer Science and Technology Division

ABSTRACT

The central scientific goal of the ARPA Image Understanding Project research program at SRI International is to investigate and develop ways in which diverse sources of knowledge may be brought to bear on the problem of interpreting images. The research is concerned with specific problems that arise in processing aerial photographs for such military applications as cartography, intelligence, weapon guidance, and targeting. A key concept is the use of a generalized digital map to guide the process of image analysis.

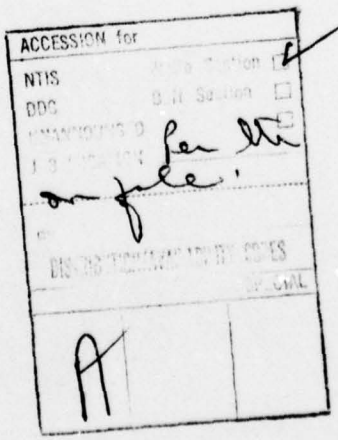
In the present phase of our program, the primary focus is on developing a "road expert," whose purpose is to monitor and interpret road events in aerial imagery. The objectives, methodology, and current status of our research are described in this report. Particular technical topics include:

(1) Data Base Construction

(2) Image-to-Map Data Base Correspondence (a detailed discussion supported by three mathematical appendices)

(3) Road Detection and Tracking

(4) Shadow and Anomaly Analysis



CONTENTS

ILLUSTRATIONS

ACKNOWLEDGMENTS

The work reported here was performed by G.J. Agin, H.G. Barrow, R.C. Bolles, M.A. Fischler, L. Quam, J.M. Tenenbaum, and H.C. Wolf.

The research was supported by the Defense Advanced Research Projects Agency of the Department of Defense and was monitored by the U.S. Army Research Office under Contract No. DAAG29-76-C-0057.

I DETECTING AND INTERPRETING ROAD EVENTS IN AERIAL IMAGERY

A. Introduction

Research at SRI International under the ARPA Image Understanding Program was initiated to investigate ways in which diverse sources of knowledge might be brought to bear on the problem of analyzing and interpreting aerial images. The initial phase of research was exploratory and identified various means for exploiting knowledge in processing aerial photographs for such military applications as cartography, intelligence, weapon guidance, and targeting. A key concept is the use of a generalized digital map to guide the process of image analysis.

The results of this earlier work were integrated in an interactive computer system called "Hawkeye" [3]. This system provides necessary basic facilities for a wide range of tasks and a framework within which specialist programs can be integrated.

Research is now focused on the development of a program capable of expert performance in a specific task domain: road monitoring. The following sections of this report present an overview as well as some recent technical results produced in this ongoing effort.

B. Objective

The primary objective of this research is to build a computer system that "understands" the nature of roads and road events. It should be capable of performing such tasks as:

* Finding roads in aerial imagery
* Distinguishing vehicles on roads from shadows, signposts, road markings, etc.
* Comparing multiple images and symbolic information pertaining to the same road segment, and deciding whether significant changes have occurred.

It should be capable of performing the above tasks even when the roads are partially occluded by clouds or terrain features, or are viewed from arbitrary angles and distances, or pass through a variety of terrains.

C. Approach

To achieve the above capabilities, we are developing two "expert" subsystems: the "Road Expert" and the "Vehicle Expert." The Road Expert knows mainly about roads, how to find them in imagery, and what things belong on them. It works at low-to-intermediate resolution (e.g., from 1 to 20 feet of ground distance per image pixel) and has the ability to distinguish vehicles from other road detail. The Vehicle Expert works on higher-resolution imagery and can identify vehicles as to type. We are concentrating our efforts on the Road Expert and therefore will limit our discussion to this component of our system.

The major tasks automatically performed by the Road Expert are:

* Image/Map Correspondence: Place a newly acquired image into geographic correspondence with the map data base.
* Road Tracking: Precisely mark the centerline of selected visible sections of road in the image.
* Anomaly Analysis: Locate and analyze anomalous objects on, and adjacent to, the road surface; identify potential vehicles.

The image/map correspondence task is accomplished by locating roads and road features as landmarks; correspondence is performed at resolutions as coarse as 20 feet/pixel so that a reasonably wide field of view (10 to 100 square miles) can be processed at one time. It is nominally assumed that the initial combinations of uncertainties about the estimates for the camera parameters implies uncertainties on the ground of approximately +/- 200 feet in X and Y. The correspondence procedure works iteratively to refine the camera parameters. A typical goal is to reduce the implied uncertainties on the ground to about +/- 2 feet in X and Y.

Having placed the image into correspondence with our map data base, one or more of the visible road sections are selected for monitoring. The road center-line and lane boundaries are found to an accuracy of one to two pixels in imagery with a resolution of 1 to 3 feet/pixel.

Given the precise road locations in the image, anomalous objects are detected by scanning on and along the road pavement. These anomalous objects are then identified as to type (e.g., vehicle, shadow, road surface marking, signpost, etc.).

The above tasks will be supported by information about road condition and general structure from a symbolic data base. For example, if prior photographic coverage of the area being analyzed is available, the problem of anomaly classification can be simplified by determining if a similarly shaped anomaly could be found in the same general location over some extended period of time. Additional examples of how data-base knowledge and stored models can aid in the analysis process include: using the time of day in discriminating shadows from objects of interest; utilizing the general shape and width of the road (obtained from a map) as an aid in road tracking; providing relevant information on the anticipated size, shape, and road orientation of potential vehicles.

A central theme of this effort is to consider roads as a knowledge domain. In particular, we are addressing the question of how a-priori knowledge can be directly invoked by the image-analysis modules (what type of knowledge, how should it be represented, and what are the mechanisms for its use). To achieve our goal of building a very-high-performance system, we are developing explicit models of the image structures we are dealing with, and additionally, models of the decision procedures embedded in the image-processing algorithms so that the algorithms can evaluate their own performance. Finally, we are planning an overall control structure which will be concerned with the problems of coordinating analysis across a spectrum of levels of resolution, and with integrating multisource information.

D. Progress

1. Data Base Construction

An underlying assumption of our overall approach is the existence of a map data base to guide the image analysis process. A significant part of our effort is thus concerned with the questions of what information this data base should contain and how it should be structured; and then assembling the needed data.

We have selected five distinct geographic sites scattered around the San Francisco Bay Area, have acquired multiple photographic coverage for each of these sites, and are currently building a detailed data base for one of these sites (PM280). Figure 1 shows one of our images of this site.

At present, the Road Expert data base contains two different forms of information. The first form is a loosely coupled collection of digital and nondigital information about our test sites. The second form is an initial implementation of a tightly integrated digital data base for each site.

The following sources of information have been used to construct the data base:

(1) digitized aerial images of the various sites including information concerning camera focal length, day of year, approximate altitude and location

(2) USGS 7.5 minute series topographic maps (the 3-D information in these maps is of very limited utility for our purposes due to the crude altitude and spatial resolution)

(3) California Department of Transportation road construction plans for some sites containing post-construction survey data

The current digital (site) data base consists of a collection of disk files containing information about linear road segments and "point" features on the road surface.

Each linear road segment is described by the 3-D coordinates of its end-points, its width, and a photometric model for the road

cross-section. Each segment description also includes a pointer to other nearby road segments whose relative positions can be used by the Road Expert for verification of acquisition.

Each point feature described in the data base is assumed to lie on a horizontal plane surface, but this restriction will be relaxed in the future. The photometric appearance of each point feature is defined by extracting a window containing the feature from some previously seen image of the site (see Figure 2). The 3-D geometry of the patch is defined by the coordinates of the window in the image, the calibration of the image to the 3-D world coordinates, and the z-elevation of the road surface at this point feature. The present structure and content of the data base was chosen in order to support experiments in automatic acquisition and calibration (see Section II of this report); consequently, it is still incomplete with respect to other needs of the road expert. One addition currently planned is to provide a more complete geometric model for the principal roads at each site. This will enable the data base to direct the road tracker to analyze an entire site automatically.

In addition to expanding the size and scope of our data base along the lines indicated above, we plan to use the capabilities of the Road Expert itself to automate many of the steps required for such data base construction.

2. Image/Data Base Correspondence

This task involves locating a few known road features (landmarks) in a newly acquired image, and then using the correspondence between the location of these landmarks and their geographic coordinates as stored in our map data base to determine the precise location (and orientation) that the "camera" was in when the image was acquired. Given the camera parameters, we can now derive a transformation that will assign geographic (x, y, z) coordinates to every point in the image. Figure 2 shows some of the landmarks we are currently using for the PM280 site. The search in the image for the landmarks is a

sequential process guided by our continually more precise estimate of the camera's location. Figure 3 shows an example of the uncertainty ellipse generated by the "camera calibration strategist" to delimit the search for the first landmark. (This ellipse is based on a mathematical model of the calibration process and assumed a-priori knowledge of initial uncertainty in camera location.) Once the first landmark has been located, the camera calibration strategist can refine position estimate and even further narrow the search for the second landmark as also shown in Figure 3.

Our work on the correspondence problem, employing an iterative approach which combines error modeling, feature matching, and refinement of the camera location estimate, has resulted in a number of extensions to the existing theory. A more complete exposition of the above approach and its status is contained in Section II and Appendices A-C. However, it is important to note here that we have been able to establish image/map correspondence to an average error of between 2-3 feet of ground distance. Thus, given the potential robustness of this approach, we believe that it can play an important role in an image matching navigation or terminal homing system (e.g., the cruise missile).

Additional work in this particular task will be primarily directed to improving the performance and flexibility of our landmark detectors, especially in regards to the question of verification and filtering out of false matches.

3. Road Tracking

We have evolved a number of techniques capable of tracking roads in aerial imagery across a 1-20 feet/pixel spectrum of resolutions. Most of these results have been described in our previous reports (see References 4 and 13) and under the conditions available in our current imagery, perform extremely well. Figure 4 shows the performance of the low-resolution road tracker. The low-resolution road tracker uses a road model which assumes local homogeneity in intensity

along the road and contrast in intensity between the road and the adjacent terrain. A linking algorithm uses an optimization technique to find a "best estimate" of the global road path based on local agreement with the road model described above. Figures 5a and 5b show some examples of the high-resolution road tracker. Using a road model in which we assume segments exhibiting relatively smooth/slow changes in direction and also in the intensity profile normal to road direction, we have been able to achieve surprisingly robust performance in tracking the road center line. In many cases, roads that have almost no discernible contrast at their edges can be readily followed. Note that the clouds appearing in these images were generated by a synthesis program we were forced to resort to in order to get a variety of cloud cover conditions needed to adequately test our techniques (see Appendix D).

Future work on road tracking will be primarily concerned with maintaining current levels of performance as the viewing conditions become increasingly more difficult (e.g., greater degrees of cloud cover or occlusion by shadows and adjacent terrain features) and with the problem of "verification." Rather than just making a best estimate of road location, we want the road tracker to also estimate the likelihood that this best estimate is indeed a visible segment of road.

4. Anomaly Analysis

The high-resolution road tracker discussed earlier assumes that roads in images are regions where the brightness varies in a predictable way. Small regions in which the brightness is significantly different from that predicted by the road model are called anomalies. These anomalies arise from such things as vehicles, road markings, shadows of various objects on or off the road, overhanging trees, and discolorations of the road surface. We are investigating methods for detecting and classifying these anomalies.

We have augmented the high resolution road tracker to produce a "difference image" obtained by subtracting the road model from the

original image. This difference image produces isolated and enhanced anomalies simplifying the following analysis and classification tasks. The initial detection of anomalies is done by thresholding the absolute value of the difference image. The optimum threshold to apply is a function of the variation to be expected in the road surface. This variation is calculated during the correlation road tracking phase as the RMS average amount by which the road surface differs from the road model, after suspected anomalies are masked out. Figures 6a through 6d show an example of the above process.

Understanding shadows in aerial images is crucial to successfully classifying the anomalies. A significant proportion of the anomalies in our library of images are shadows of objects. The vehicles themselves cast shadows, which must be removed from the initially-detected anomaly before classification can take place. Even more to the point, shadows can serve a useful purpose in helping to locate vehicles, and can also be used as landmarks in performing the correspondence task.

In addition to finding shadows as deviations from the detected road model, we are investigating two additional techniques which appear rather promising. First, we note that shadows are usually among the darkest objects in an image. If we can properly select an intensity threshold, we can mark the shadows (at least on the road surface) and exclude almost everything else. Local threshold setting can be accomplished by choosing a value lower than the measured intensity of some known dark area on the road surface, such as a tar patch (located using map data base information), or even the oil slick which appears in the center of each lane of almost any road. On the other hand, the threshold should not be set lower than the measured intensity of shadows either detected in the image, or predicted from data base information. Figures 7a and 7b show some examples of the effectiveness of this form of threshold-based shadow detection.

A second approach to detecting shadows is based on the fact that for the locally planar and constant reflectance road surface (at least along a path parallel to the road direction) the intensity

variation across a shadow edge is a function of the ratio of secondary (diffuse sky-light) to primary (direct sunlight plus diffuse sky-light) illumination, and is roughly constant in any single image. Once we have found one shadow (e.g., by predicting its location from data base information, time of day, date, and latitude and longitude of the scene) we can determine the required ratio (or intensity difference in an image digitized on a logarithmic brightness scale) and use it to detect other shadows on the road surface. Obviously the ratio will have some range of variation, and in particular it will be somewhat higher for shadows cast by small or thin objects (such as passenger cars) than for shadows cast by large, solid objects (such as, say, a freeway overpass). The ratio for a shadow edge falling across a road oil slick might also tend to be a bit higher than the ratio for a clean section of pavement because of reduced film sensitivity in the darker area. Figure 8 shows some typical examples of the intensity ratio across shadow and non-shadow edges on the road surface in an image.

The problem of distinguishing vehicles from other road anomalies can be simplified by noting that, in addition to their size and shape characteristics, vehicles have a range of local intensity variations (due to shadows, highlights from metal and glass, differently oriented surfaces, etc.) far exceeding that of most other road artifacts. Once a vehicle has been detected, additional analysis usually requires separating the image from its cast shadow. This can be accomplished in a number of different ways. For example, we can use the methods for general shadow detection mentioned above, or we can predict the location of the shadow by assuming the vehicle height (five feet for for passenger cars) and knowing the sun location, time of day, etc.

Another technique for separating a vehicle from its shadow is based on the specific assumption that vehicles are likely to be rectangular in their aerial views, with their long edges oriented parallel to the road. If pixels in either the difference image or the original are projected to a line perpendicular to the road orientation and we plot average brightness as a function of distance across the

road, we see a significant discontinuity at the boundary between car and shadow.

Our work in the next several months will concentrate on combining the results of several different tests to determine not only what is and is not shadow, but also to actually classify each anomaly. We hope the method will be general enough to accommodate various kinds of evidence. It should take into account each method's estimate of its own confidence, if it can be obtained. Rather than choose one method over another, we hope to be able to integrate the results to come up with a consensus.

E. Comments

We see the military relevance of our work extending well beyond the specific road monitoring scenario presented above. In particular, a Road Expert can be applied to such problems as:

(1) Intelligence: monitoring roads for movement of military forces

(2) Weapon Guidance: use of roads as landmarks for "Map-Matching" systems

(3) Targeting: detection of vehicles for interdiction of road traffic

(4) Cartography: compilation and updating of maps with respect to roads and other linear features

In accord with our generalized view of the applicability of the Road Expert we are constructing, we are attempting to achieve a level of performance and understanding in each of the functional tasks which far exceeds that which would be required for dealing with the road monitoring scenario alone.

The remainder of this report presents a detailed discussion of our image-to-map data base calibration procedure (supported by three mathematical appendices).

II THE SRI ROAD EXPERT: IMAGE-TO-DATA BASE CORRESPONDENCE

A. Introduction

Computing an image-to-data base correspondence is a general problem occurring in all knowledge-based systems. In most image tasks the correspondence is a projective transformation and can be modeled as a function of the camera parameters, such as focal length, X, Y, Z, heading, pitch, and roll. If the parameters are known precisely, the model can precisely predict the two-dimensional image coordinates for any three-dimensional data base point.

One common form of the image-to-data base correspondence problem is to be given good estimates of the camera parameters and be asked to improve them. This task is important in many military situations. For example, in navigation it is the crucial step that improves the system's estimate of the location of the plane or missile. In change detection it is used to align two images of the same area so that the corresponding regions can be compared. In the Road Expert it is the key to the utilization of the data base in subsequent tasks such as road monitoring.

The basic approach we are using to refine a correspondence is to locate known features in the image and use their locations to improve the correspondence (see Figure 9). The data base contains descriptions of the available features. From these descriptions a set of features is chosen to be located that is based on the predicted viewpoint and viewing conditions. The estimates of the camera parameters are used to predict what the features look like and where they are likely to appear. Feature detection techniques ("operators") are chosen to locate the features and they are applied. Since the operators may not locate their intended features, their results are verified either by locating a larger portion of the features or by checking the relative positions of

other features. After a set of features has been found, their locations are used to refine the estimates of the camera parameters. The parameters are refined by searching the parameter space for sets of parameter values that minimize the distances between the predicted locations of features and the locations determined by the operators. If the correspondence is not precise enough, the whole process can be repeated.

The important computations and decisions required to refine a correspondence are listed below:

(1) selection of features

(2) prediction of the appearance of a feature

(3) selection of an operator to locate the feature

(4) prediction of the nominal image location of a feature

(5) prediction of the range of image locations about a feature's nominal location

(6) selection of the order in which to apply the operators

(7) application of the operators

(8) verification of the results produced by an operator

(9) decision of when to use the results of one or more operators to help other operators locate their features

(10) decision of when to update the whole correspondence

(11) computation of a refined correspondence

(12) decision to stop

A number of people have worked on individual items in this list [1, 5, 6, 7, 8, 9, 10, 11, and 12], but mainly for pairs of images that were taken closely in time and from similar viewpoints.

There are several factors in the military domain, as well as other domains, that increase the difficulty of these items beyond current capabilities. Examples of such factors are a wide variety of viewpoints, a distribution of shadows, and the possibility of clouds. All of them make it more difficult to select features, predict the appearance of features, and locate features. Therefore, they increase the need for feature verification and strategy-based decisions. Which

operators should be used for an image taken from this viewpoint and under these conditions? When should the results of one operator be used to reduce the predicted search area for a nearby feature? This type of question becomes more important as features become harder to find.

Our research goal is to produce an automatic system to refine correspondences within the road domain. To reach this goal we need to develop new models and techniques for several of the items in the above list. So far we have concentrated on a few of them: the prediction of the range of image locations for a feature, the verification of the results of an operator, and the computation of a refined correspondence. In this section we will state our assumptions, describe our new techniques, and present an example.

B. Assumptions

Our assumptions are summarized in Figure 10.

Figure 11 is a typical picture to be processed by the system. We assume that the resolution of the digital images will be between 20 feet/pixel and 1 foot/pixel. Figure 12, which is another picture of the site shown in Figure 11, is displayed so that one pixel corresponds to approximately sixteen feet on the ground. Figure 13 is a portion of Figure 11 displayed at its full resolution of approximately 1 foot/pixel.

We assume that we will have a data base of the area on the ground contained in each picture to be analyzed. The data base contains the geometry and topology of the roads and the locations of other features, such as road markings. Since we expect to obtain repetitive coverage of the areas of interest, the data base may also contain information about the appearances of the road sections and features derived from previous images.

Images of the same site may be taken at different times of the day so the shadows may be different. Notice the variation in shadows between Figures 11 and 12. Part of the information expected by the

system for each picture is the day of the year and the time of day at which the picture was taken.

Some of the images may contain clouds that obscure some of the roads and other data base features (e.g., see Figure 14); and more generally, terrain features, buildings, and trees may obscure features of interest. The implication is that the system should be able to handle operators that find multiple matches, incorrect matches, or no matches at all.

Different pictures of the same region may be from different viewpoints. In particular, they may be from significantly different altitudes (e.g., twice as high) or different angles (e.g., 45-degree obliques versus vertical pictures). Figures 11 and 12 are pictures of the same site except that Figure 12 was taken from approximately twice the height and at a heading that is different from that of Figure 11 by almost 90 degrees. The wide variety of viewpoints implies that intensity correlation is not always sufficient to locate features. Other operators will be necessary.

Even though the viewpoint may vary widely, we expect to be given good estimates of the camera parameters for each picture. The camera parameters can be factored into two convenient sets: internal camera parameters and external camera parameters. The internal parameters describe the camera-specific information, such as the focal length of the lens. The external parameters describe the relative position and orientation of the camera with respect to the world represented in the data base. Generally, the a priori estimates of the internal parameters are much better than the estimates of the external parameters.

We expect a measure of the uncertainty associated with each parameter estimate. For example, the HEADING might be estimated to be 75 degrees, plus or minus one degree. These uncertainties are used to predict the regions in a picture to be searched in order to locate a feature. We will refer to these search regions as "uncertainty regions." The smaller the uncertainties, the smaller the uncertainty regions; the smaller the uncertainty regions, the easier it is to automatically locate the desired features.

Two of our most important assumptions restrict the range of initial uncertainties about the camera parameter estimates. The first one restricts the combined internal and external uncertainties so that they do not imply uncertainty regions on the ground of more than approximately plus or minus 200 feet. The second one restricts the size of each parameter's uncertainty so that it is relatively small. The first assumption, in effect, restricts the sizes of the uncertainty regions that have to be searched to locate a feature. For example, if an image has a resolution of 1 foot/pixel, the largest uncertainty region would then be approximately 400 x 400 pixels. The second assumption limits the portion of the parameter space that the optimizer has to search. It also indirectly limits the maximum geometric change in the appearance of a feature.

An implicit assumption behind the characterization of a correspondence as a function of the camera parameters is that the imaging process can be modeled as a perspective transformation. If it cannot, a different mapping function would have to be used, but the same numerical approach would apply.

C. Uncertainty Regions

Given parameter estimates and uncertainties about those estimates, where in the image is a feature likely to appear? Or more specifically, what region in the picture will have a given probability (e.g., a 95% probability) of containing the feature? To answer this question, one has to predict the effect on the location in the image of a feature caused by changing the parameter values in accordance with their stated uncertainties. To do that, one needs a model of their uncertainties. The error model we use is that the parameters vary according to a joint normal distribution, which is a reasonable assumption for measurements produced by a device such as an inertial guidance system because each parameter's error is a sum of several small errors. For this model the uncertainty regions are ellipses in the image plane. The derivation of this fact can be found in Appendix A.

Figure 15 shows a typical uncertainty ellipse that is prescribed to have a 95% probability of containing the actual occurrence of the feature. The 100 dots were produced by varying the camera parameters 100 different times according to the error model and by projecting the three-dimensional feature point onto the image plane containing the ellipse. Notice that 92 of the points are inside the ellipse, which is consistent with the 95% prediction.

Having found one feature, one would expect that its location would greatly restrict the possible locations for a nearby feature. This idea leads to a second type of uncertainty region, a relative uncertainty region. In addition to the normal information used to compute an uncertainty region, a relative uncertainty region is a function of another feature and its location. Since the location of a nearby feature typically adds constraints on the possible locations for a feature, the relative uncertainty region is usually significantly smaller than the regular uncertainty region. Given the assumption that the camera parameters vary according to a joint normal distribution, the relative uncertainty regions are also ellipses. A derivation of the mathematical description of a relative uncertainty region is given in Appendix B.

A relative uncertainty region is used to reduce the amount of work required to locate a second feature after a nearby feature has been found. This is particularly useful when a possible match for a feature is being verified. The logic is as follows: if this is feature A, then feature B should be in a small region over there; if B is not there (and not occluded), this must not be A.

Figure 16 shows the initial uncertainty ellipse and the relative uncertainty ellipse about a point feature. The large ellipse is the uncertainty region predicted from the uncertainties about the camera parameters. The small ellipse is the relative uncertainty region derived from the location of the arrow just above it in the picture.

D. Point-on-a-Line Matches

Almost all previous work has involved the use of point-to-point matches to refine correspondences. Since roads are the major objects of interest for the road expert, we wanted to include them as features that could be used within the image-to-data base correspondence phase as well as in the monitoring phase.

There is a built-in trade-off between point features and line features, such as roads: it is easier to find a point on a line than it is to locate a point feature, but less information is gained by doing so. Point-to-point matches produce twice the number of constraints for the refinement process, but they are generally more expensive to find because an area search is required as opposed to a linear search for point-on-a-line matches.

To use linear features we needed an operator (or operators) to find points on roads and we had to to extend the correspondence refinement process to include the new type of feature match.

1. Point-on-a-Line Operators

Currently we have two operators that locate points on a road. One is used at low resolution (e.g., 20 foot/pixel) when roads appear as lines, and one is used at high resolution (e.g., 1 foot/pixel) when the internal structure of the road is discernable. The low-resolution operator is an extension of the Duda road operator, which has been discussed in previous SRI image-understanding reports [2]. The high-resolution operator is an adaptation of Quam's road tracking operator [12]. It performs a 1-D correlation of the expected road cross section to locate possible points on the road and then tries to track the road for a short distance to make sure that the candidate point is part of the expected road.

2. Correspondence Refinement

The correspondence refinement process (or "optimizer") is based on Gennery's approach to calibration [11] (see Appendix C). It solves the nonlinear problem by iteratively solving linear approximations. For point-to-point matches a 3-D point in the world is matched with a 2-D point in the image. In that case the optimizer has two residuals per match to use to improve the camera parameter estimates: the X and Y components of the difference between the predicted image of the world point and the point in the image at which the operator located its match. If instead of locating a specific point, an operator locates a point on a line, the optimizer only has one residual to use because the point could be any place along the line. The residual for a point-on-a-line match is the distance from the point to the line. As the optimizer searches for improved camera parameters, the image of the 3-D line should get closer to the point located by the operator, but the closest point on the line may slip back and forth along the line.

So far the optimizer has only been extended to handle point-on-a-line matches. However, since roads are generally constructed as combinations of linear segments and arcs of circles, it may be useful to extend the optimizer to include other types of matches that involve a point and an analytic curve, e.g., a point-on-an-ellipse match. The main components of such an extension are (1) a procedure to compute the distance between a point and the curve and (2) a procedure to compute the partial derivatives of that distance with respect to the camera parameters.

The optimizer could even be extended to arbitrary curves by incorporating a procedure, such as chamfering [5], that computes the distance between a point and an arbitrary curve.

The current implementation of the optimizer is relatively fast. It takes one second on our KL-10 to perform one iteration when 100 residuals are used to refine the estimates. (Recall that each point-to-point match adds two residuals; each point-on-a-line match adds

one residual.) Five to ten iterations are normally required to achieve convergence, which is defined to be a state in which the parameter adjustments are on the order of .00005 units.

As Gennery points out, the optimizer can be used to filter out "mistakes" by iteratively deleting the match with the largest residual until the deletion no longer significantly improves that point's residual. In practice this heuristic has proven to be useful, but it is expensive and theoretically unsound. For example, consider Figure 17, which shows a set of points through which a line is to be fitted using a least-squares approach. The one "mistake" happens to draw the line toward it in such a way that the point with the worst residual after convergence is one of the "good" points. Deleting the point with the worst residual and trying again only repeats the situation. The conclusion is to try to filter out mistakes before they are given to the optimizer. The next subsection describes some of the ways this filtering or verification can be done.

E. Feature Verification

As mentioned in the last subsection, it appears to be more cost-effective to filter out mistakes, if at all possible, before applying the optimizer. We have identified four possible methods for performing such filtering:

(1) Operator threshold--Be suspicious of any match for which the operator does not produce a confidence above a certain threshold; e.g., if a 2-D correlation operator produces a correlation of less than .8, ignore its results.

(2) Self-support--Be suspicious of any match that cannot be verified by locating a larger portion of the same feature; e.g., if an operator locates a point that is supposed to be on a road but the road tracker cannot extend the match, ignore it.

(3) Pairwise support--Be suspicious of any match that is not positioned correctly relative to some other feature that has already been located; e.g., if an operator locates an arrow on a road and its matching location is not at a reasonable distance from another nearby feature that has been verified, ignore the match.

(4) Group support--Be suspicious of any match that is not positioned correctly relative to a group of other features that have already been located, e.g., if three point features have been found and verified, ignore a match for a fourth feature that does not appear at the correct relative location.

We differentiate between these methods (or heuristics) because they generally require different models and techniques.

It is relatively straightforward to apply all of the verification methods to point features. The relative uncertainty regions can be used to determine if two features are mutually consistent. This pairwise consistency can be extended to group consistency through maximal clique techniques [1] or through optimal embedding techniques [9].

The extension to group consistency can be achieved by constructing a graph that has one node for each match and a link between each pair of nodes that is pairwise consistent. The largest completely connected subgraph (i.e., the largest maximal clique) represents the largest set of mutually consistent matches. Any match that is not in that set is pairwise inconsistent with at least one of the matches in the set. Thus, it is suspicious.

Additional care has to be taken to apply the verification techniques to point-on-a-line matches. The important test is to be able to distinguish pairwise consistent matches from pairwise inconsistent matches when one or more of the matches is a point-on-a-line match. Figure 18 shows the three significantly different cases. In Figure 18a one of the two matches is a point-to-point match and one is a point-on-a-line match. If the slope of the line is known accurately, the distance between the point and the line can be used to determine if the matches are consistent. Since the uncertainties associated with each camera parameter are relatively small, the slope of the line should remain relatively constant. Thus the distance from the point to the line should be relatively constant.

In Figure 18b both of the matches are point-on-a-line matches, and the lines are essentially parallel. In this case the distance between

the lines is sufficient to check the relative positions of the two matches. For example, if an operator is trying to locate both sets of lanes on a freeway, the distance between the two sets of lanes should be within a predetermined range.

If both of the matches are point-on-a-line matches and the lines are not parallel, as in Figure 18c, some additional information is needed in order to check their relative consistency. One solution is to intersect the two lines and use that point in conjunction with a third match to check the relative position of all three matches.

F. Example

We have implemented one fixed strategy in terms of the verification techniques and are just beginning to explore the possibility of automatically tailoring the verification strategies to fit specific sets of features and tasks. The example task is to refine the image-to-data base correspondence for the picture shown in Figure 12 using its full resolution of approximately 2 feet/pixel. The initial uncertainties about the camera parameters imply uncertainties in the image of plus or minus 95 pixels, which correspond to approximately plus or minus 190 feet on the ground. The goal is to reduce these uncertainties to approximately plus or minus one pixel, an increase in precision of almost two orders of magnitude.

The data base used in this example contains two types of features, linear road segments and road surface markings. Figure 19 shows the locations of features that are available for this site. The lines represent the road segments and the pluses represent the surface markings. The appearance of each road segment is described by a road cross section model. The appearance of a surface marking is described by an image patch from a previous picture of the site.

A fixed strategy has been implemented to use these features to perform the task and demonstrate our new techniques. The basic approach is to locate the linear features first because they are less expensive to find, use them to refine the camera parameters, locate the point

features, use them to verify the first refinement, and then perform a second refinement using both the points and the lines.

Given estimates for the camera parameters, the system predicts the location of the road segments in the new picture. Figure 20 shows these predictions, which are shifted left and down approximately 60 pixels from their actual locations. The estimates of the camera parameters are also used to warp each road cross section to the expected size and orientation of the corresponding road segment. In addition, the estimates of the uncertainties about the camera parameters are used to predict the uncertainty regions about the center points of each linear segment. Figure 21 shows those uncertainty ellipses that have a 95% probability of containing the desired point.

The search strategy for a linear feature is to look along lines perpendicular to the expected location of the feature. The lengths of the lines are determined by the size of the uncertainty ellipse.

The high-resolution, one-dimensional correlation operator is applied along the search line to locate points that may be on the desired road. The self-support method is used to verify each candidate point. The road tracker tries to track the road for a short distance. If it cannot, the point is abandoned. Figure 22 shows an example of the application of self-support. The line on the left is the predicted location of the road segment. The other line, which is crossed like a T, represents the location of the match and the results of the road tracker following the road.

For some road segments self-support is not sufficient to locate the desired road because there are two or three parallel roads that all look alike. In order to distinguish one road from another, preplanned groups of features have been established within which pairwise and group support can be obtained. For example, Figure 23 shows a set of three sets of lanes, two of which are difficult to tell apart simply by looking at their road cross sections. The relative locations of the three sets of lanes are used to determine the correct matches. The lines perpendicular to the roads indicate the final choice for a consistent set of matches.

Figure 24 shows the results of searching for all of the road segments in the data base (shown in Figure 19). Two of the roads were not found because the contrasts were not sufficient to produce matches with the desired confidence. The matches were given to the optimizer along with the initial estimates of the camera parameters and the uncertainties about the estimates; the optimizer produced new estimates for the parameters and new uncertainties. Figure 25 shows the new predictions for the locations of the road segments. The new uncertainties imply uncertainties in the image of approximately plus or minus 1.5 pixels, close to our goal.

To verify the new estimates the surface markings were located. The new estimates were used to predict the locations and appearances of the features; the new uncertainties were used to predict the uncertainty regions; and two-dimensional correlation was used to locate the features. The average difference between the predicted location and the matching location was approximately 1.3 pixels, and the largest distance was 1.7 pixels. The final refinement based on both the lines and the points reduced the uncertainties in the image to approximately 1.1 pixels, which is very close to our goal and corresponds to approximately 2.2 feet on the ground.

We have begun to experiment with pictures containing clouds that obscure some of the features to be used for calibration. For example, consider Figure 26 in which several of the road segments are partially occluded. Figure 27 shows the linear features that the system could find and verify.

G. Discussion

We have described and demonstrated a set of techniques to perform some of the subtasks required in an automatic system to refine image-to-data base correspondences. In particular, we discussed techniques to compute uncertainty regions, techniques to incorporate point-on-a-line matches, and techniques to verify the results of operators. These techniques were combined to form a strategy, which we demonstrated in an example task.

Additional research is required on several other key subtasks required in an automatic system; for example, the selection of features and the tailoring of a strategy to different tasks. Other needs include better feature modeling, better operators to locate features over a wide range of viewing angles and conditions, and an alternative to least-squares optimization.

REFERENCES

1. A.P. Ambler et al., "A Versatile Computer-Controlled Assembly System," Proc. Third IJCAI, pp. 298-307 (August 1973).

2. H.G. Barrow, "Interactive Aids for Cartography and Photo Interpretation," Semiannual Technical Report, SRI Project 5300, SRI International, Menlo Park, California (November 1976).

3. H.G. Barrow, "Interactive Aids for Cartography and Photo Interpretation," Semiannual Technical Report, SRI Project 5300, SRI International, Menlo Park, California (October 1977).

4. H.G. Barrow et al., "Interactive Aids for Cartography and Photo Interpretation: Progress Report, October 1977", PROCEEDINGS: IMAGE UNDERSTANDING WORKSHOP", October 1977, pp. 111-127.

5. H.G. Barrow et al., "Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching," Proc. Fifth IJCAI, pp. 659-663 (August 1977).

6. R.C. Bolles, "Verification Vision for Programmable Assembly," Proc. Fifth IJCAI, pp. 569-575 (August 1977).

7. B.L. Bullock et al., "Finding Structure in Outdoor Scenes," Hughes Research Report 498, (July 1976).

8. L. Davis, "Shape Matching Using Relaxation Techniques," TR-480, Computer Science Dept., University of Maryland, College Park, Maryland (September 1976).

9. M. Fischler and R. Elschlager, "The Representation and Matching of Pictorial Structures," IEEE Trans. Comp., No. 22, pp. 67-92 (1973).

10. T.D. Garvey, "Perceptual Strategies for Purposive Vision," Technical Note 117, SRI International, Menlo Park, California (September 1976).

11. D.B. Gennery, "A Stereo Vision System for an Autonomous Vehicle," Proc. Fifth IJCAI pp. 576-582 (August 1977).

12. B.K.P. Horn and B.L. Bachman, "Using Synthetic Images to Register Real Images with Surface Models," Proc. Image Understanding Workshop, pp. 75-95 (October 1977).

13. L.H. Quam, "Road Tracking and Anomaly Detection in Aerial Imagery," Proc. Image Understanding Workshop, pp. 51-55 (May 1978).

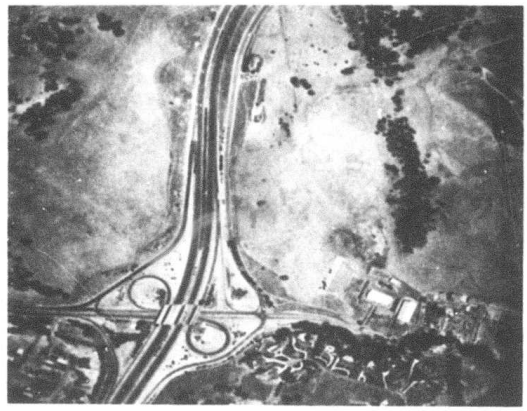


FIGURE 1 OVERVIEW OF THE PM280 SITE

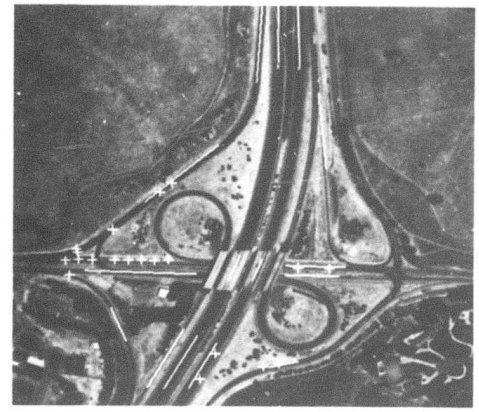


FIGURE 2(a) LOCATION OF PM280 SITE LANDMARKS

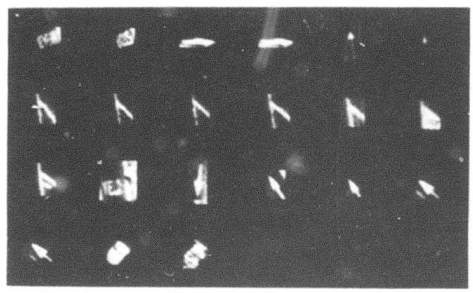


FIGURE 2(b) ROAD SURFACE MARKINGS USED AS "POINT" LANDMARKS

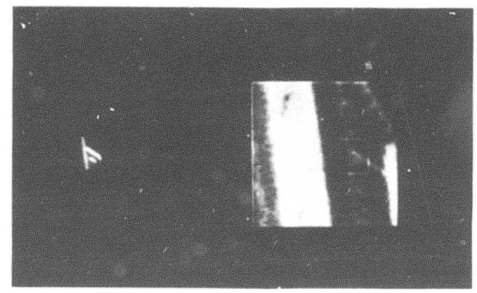


FIGURE 2(c) A POINT LANDMARK AND ITS APPEARANCE IN AN IMAGE

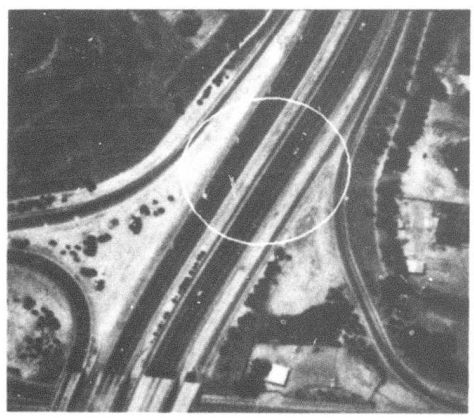


FIGURE 3 UNCERTAINTY ELLIPSES FOR LOCATING A KNOWN LANDMARK

The Larger Ellipse Represents the Initial Uncertainty in Locating a Road Surface Landmark. The Small Ellipse is the Refined Estimate of Location after One Other Nearby Landmark Has Been Located.

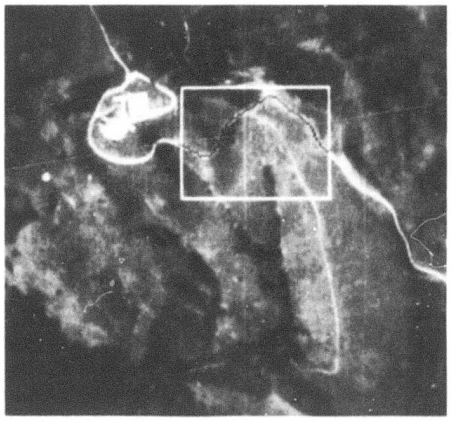


FIGURE 4 A ROAD LOCATED AND MARKED IN A SPECIFIED SEARCH WINDOW BY THE LOW RESOLUTION ROAD TRACKER

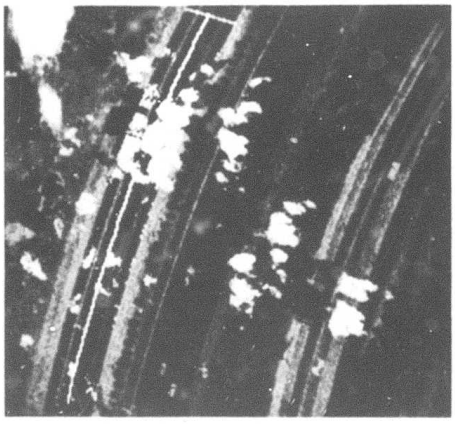


FIGURE 5(a) THE HIGH RESOLUTION ROAD TRACKER FOLLOWING A ROAD IN THE PRESENCE OF CLOUD COVER

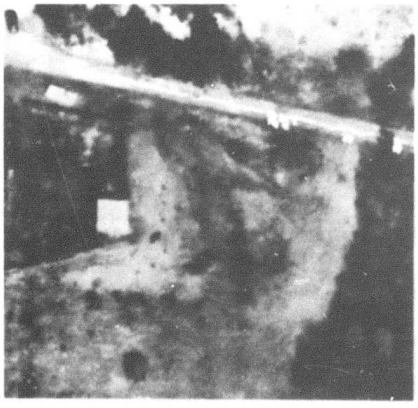


FIGURE 5(b) THE HIGH RESOLUTION ROAD TRACKER FOLLOWING A DIRT ROAD

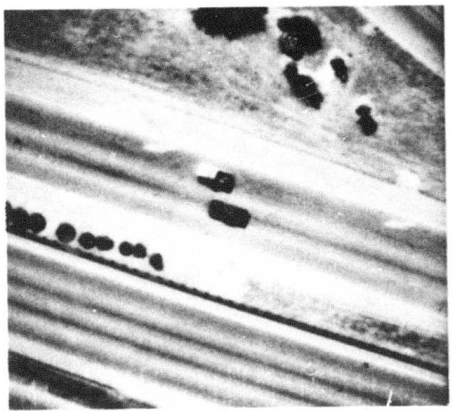


FIGURE 6(a) ORIGINAL SEGMENT OF AN IMAGE

FIGURE 6(b) DETECTION OF ANOMALOUS AREAS ON THE ROAD SURFACE

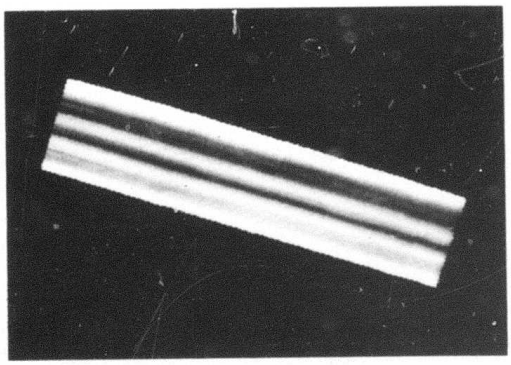


FIGURE 6(c) INTENSITY MODEL OF THE ROAD SURFACE

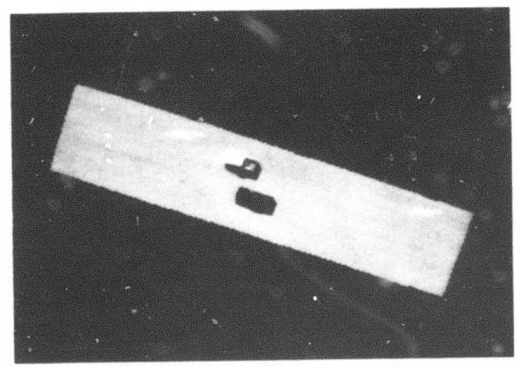
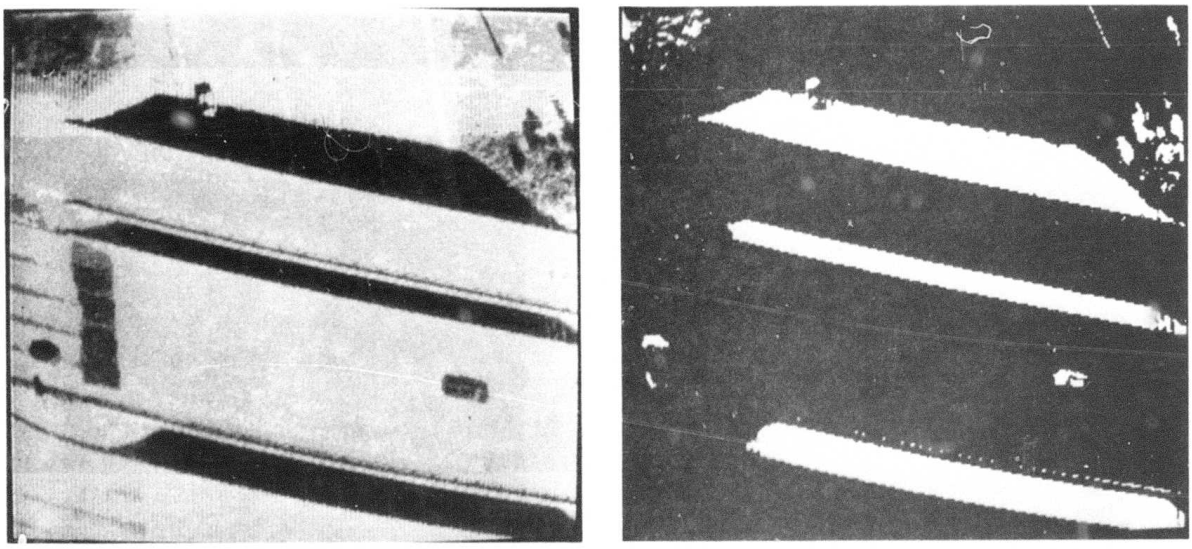
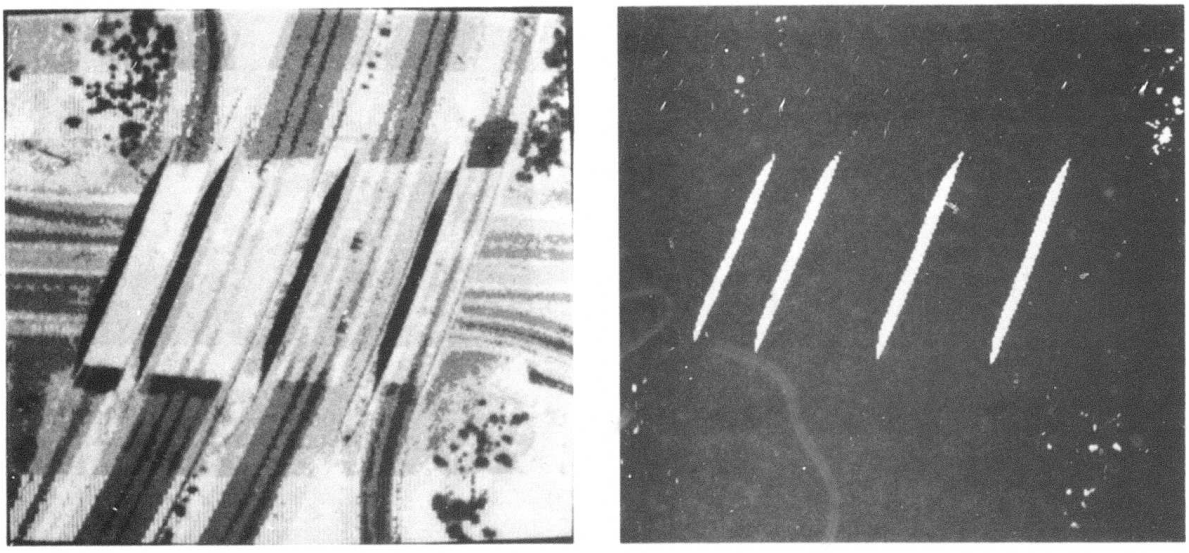


FIGURE 6(d) SUBTRACTION OF NOMINAL ROAD SURFACE INTENSITIES TO ENHANCE ANOMALIES FOR FURTHER ANALYSIS



TAR PATCH = DARK VERTICAL PATCH AT LEFT CENTER



TAR PATCH = DARK PATCH ABOVE RIGHTMOST OVERPASS

FIGURE 7(a) SHADOW EXTRACTION — THRESHOLD SET TO VALUE BELOW INTENSITY OF TAR PATCH

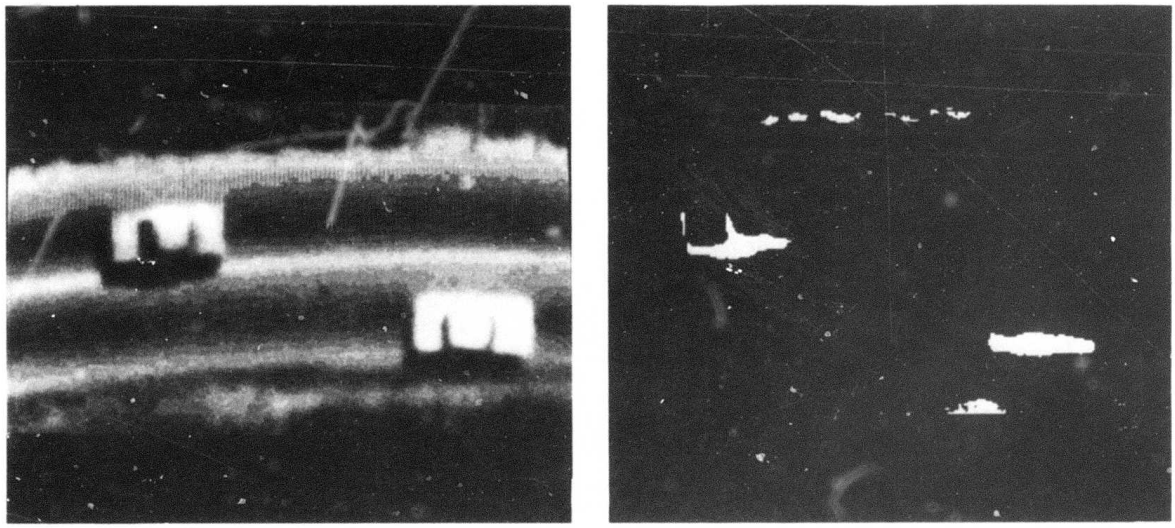
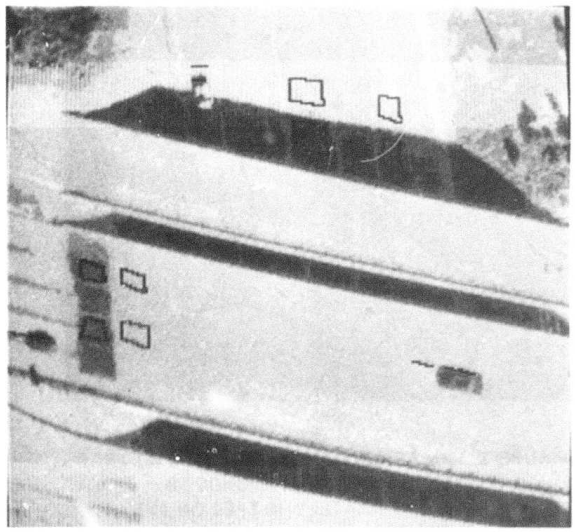


FIGURE 7(b) SHADOW EXTRACTION — THRESHOLD SET TO VALUE BELOW INTENSITY OF OIL SLICK IN THE MIDDLE OF UPPER LANE



OVERPASS SHADOWS (2 PATCHES)	0.74, 0.73
CAR SHADOWS (3 CARS)	0.79, 0.79, 0.80
TAR PATCH (2 PATCHES)	0.85, 0.86

FIGURE 8 SHADOW BOUNDARY INTENSITY RATIOS

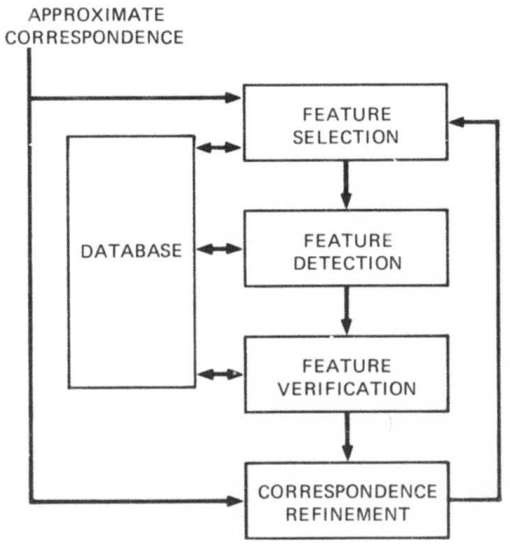


FIGURE 9 THE BASIC CORRESPONDENCE REFINEMENT PROCESS

GENERAL ASSUMPTIONS

(1) Road pictures
(2) Repetitive coverage
(3) Ground resolutions between 20 feet/pixel and 1 foot/pixel
(4) Database of roads and other features
(5) Different sun angles
(6) Database features may be obscured by clouds, terrain features, etc.
(7) Wide range of viewpoints
(8) Correspondence is a perspective transformation
(9) Small parameter uncertainties
(10) Maximum uncertainty regions on the ground of +-200 feet

INFORMATION FOR EACH IMAGE

(1) Internal camera parameters (estimates & uncertainties)
(2) External camera parameters (estimates & uncertainties)
(3) time of day and day of year image was taken

FIGURE 10 THE CORRESPONDENCE TASK ASSUMPTIONS

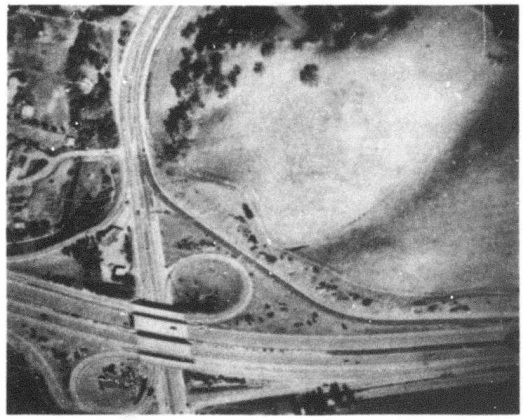


FIGURE 11 A TYPICAL AERIAL IMAGE TO BE CALIBRATED

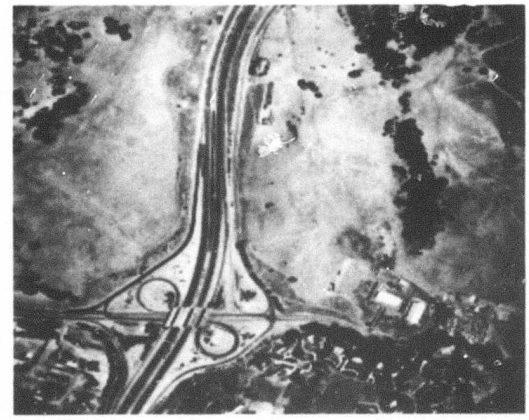


FIGURE 12 AN AERIAL IMAGE DISPLAYED SO THAT EACH PIXEL CORRESPONDS TO APPROXIMATELY 16 FEET ON THE GROUND

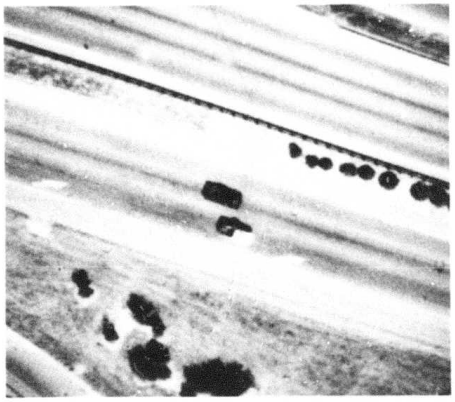


FIGURE 13 AN AERIAL IMAGE DISPLAYED SO THAT EACH PIXEL CORRESPONDS TO APPROXIMATELY 1 FOOT ON THE GROUND

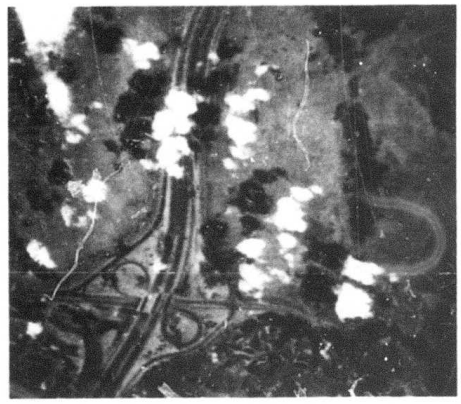


FIGURE 14 A TYPICAL IMAGE CONTAINING CLOUDS

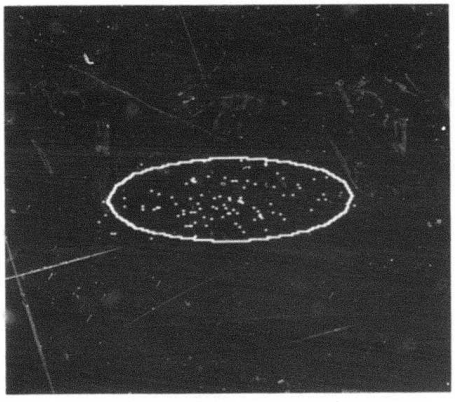


FIGURE 15 A PREDICTED UNCERTAINTY ELLIPSE AND A RANDOM DISTRIBUTION OF POSSIBLE LOCATIONS FOR THE FEATURE

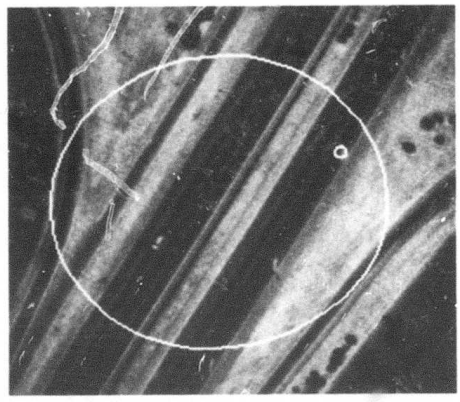


FIGURE 16 AN INITIAL UNCERTAINTY ELLIPSE AND A SMALL RELATIVE UNCERTAINTY ELLIPSE ABOUT A POINT FEATURE

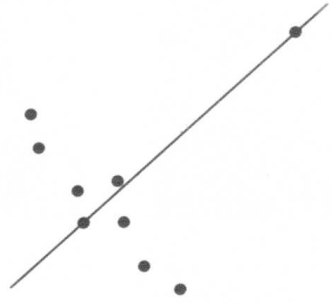


FIGURE 17 A PATHOLOGICAL EXAMPLE OF LEAST-SQUARES LINE FITTING

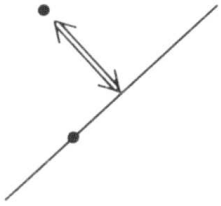


FIGURE 18(a) A POINT AND A LINE

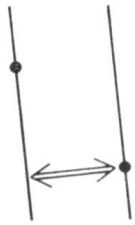


FIGURE 18(b) TWO PARALLEL LINES

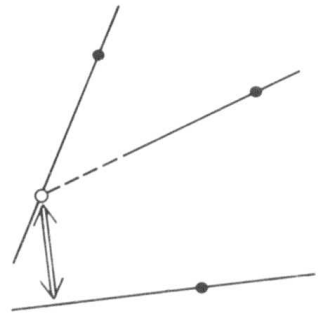


FIGURE 18(c) NON-PARALLEL LINES

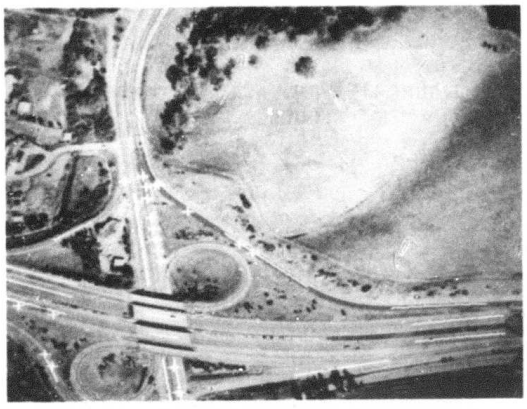


FIGURE 19 A REFERENCE IMAGE OF THE SITE AND THE LOCATIONS OF THE POINT AND LINE FEATURES TO BE USED IN THE CALIBRATION

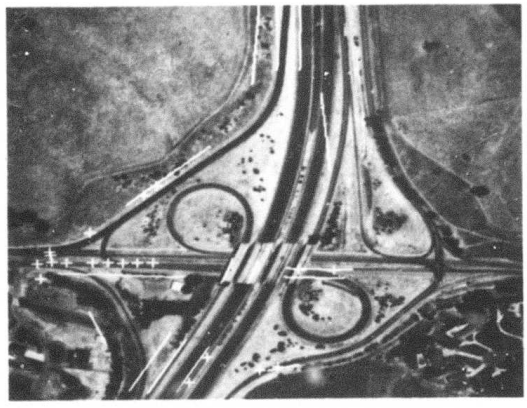


FIGURE 20 THE IMAGE TO BE CALIBRATED AND THE PREDICTED LOCATIONS OF THE FEATURES

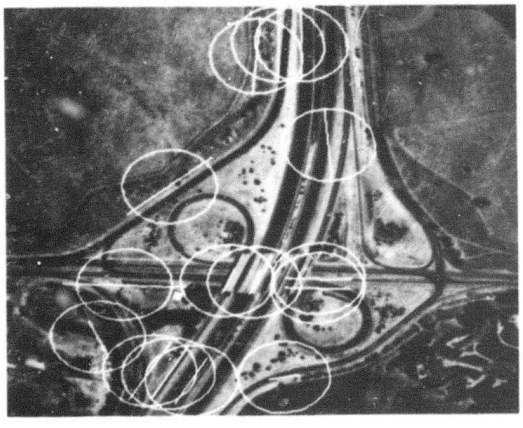


FIGURE 21 THE PREDICTED LOCATIONS OF THE ROAD SEGMENTS AND THE INITIAL UNCERTAINTY ELLIPSES ABOUT THEIR MID-POINTS

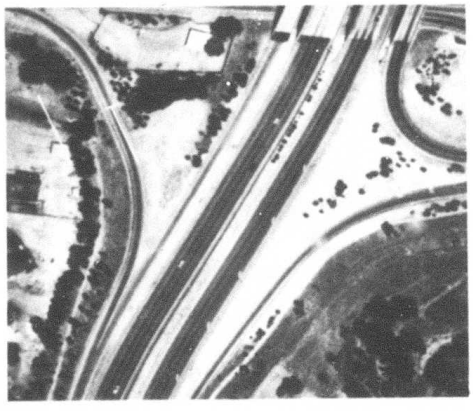


FIGURE 22 THE PREDICTED LOCATION OF A ROAD SEGMENT AND ITS MATCHING LOCATION

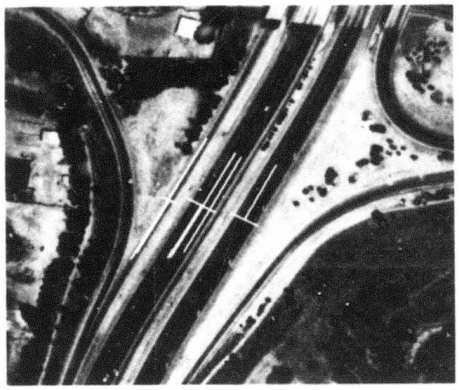


FIGURE 23 THE PREDICTED AND MATCHING LOCATIONS OF THREE ROAD SEGMENTS THAT ARE USED AS MUTUAL SUPPORT FOR EACH OTHER

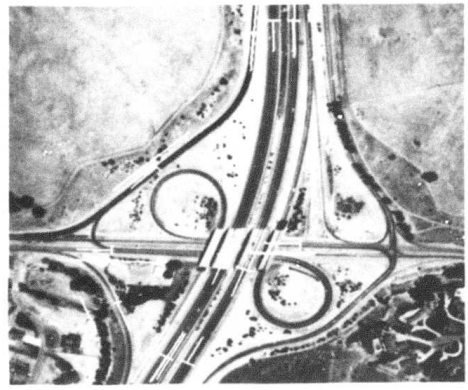


FIGURE 24 THE RESULTS OF ALL OF THE ROAD SEGMENT DETECTION OPERATORS

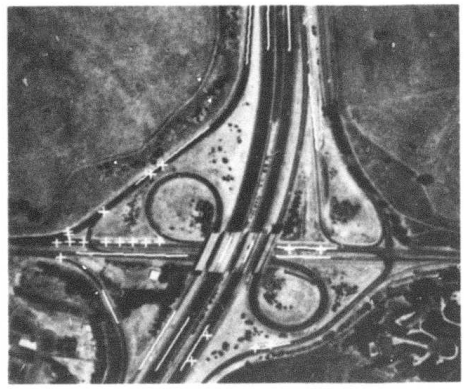


FIGURE 25 THE PREDICTED LOCATIONS OF THE FEATURES PRODUCED BY THE IMPROVED CAMERA PARAMETERS

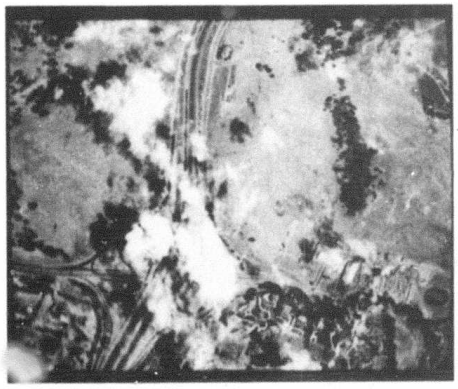


FIGURE 26 AN IMAGE TO BE CALIBRATED THAT CONTAINS CLOUDS

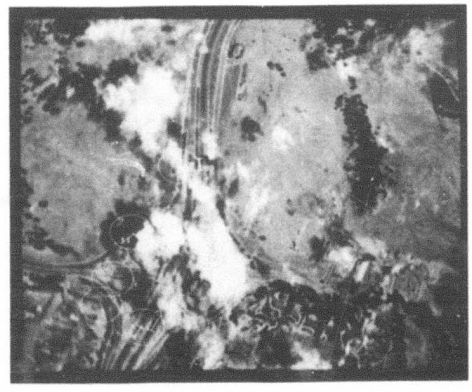


FIGURE 27 THE RESULTS OF ALL THE ROAD SEGMENT DETECTION OPERATORS

Appendix A

A LINEAR MODEL FOR PREDICTING THE DISTRIBUTION OF ERRORS UNDER A PROJECTIVE TRANSFORMATION

Appendix A

A LINEAR MODEL FOR PREDICTING THE DISTRIBUTION OF ERRORS UNDER A PROJECTIVE TRANSFORMATION

1. Problem Statement

GIVEN the set of camera parameters {yi} which define a projective transformation from 3-space to a 2-dimensional image plane {xi}, i=1,2; and assuming that the {yi}, i=1,2,...J, are jointly distributed according to a multivariate normal distribution function with given covariance matrix M, THEN we wish to find a region in the image plane, centered about the point provided by the projective transformation H{yi}, which will be large enough to contain the image of the corresponding 3-space point to some given level of probability.

2. Linear Approximation

As an approximation to the way in which the errors in the camera parameters produce displacements of a projected point, we will assume that:

$$\Delta x_1 = \sum_j \left(\frac{\delta x_1}{\delta y_j} * \Delta y_j \right)$$

[1] and

$$\Delta x_2 = \sum_j \left(\frac{\delta x_2}{\delta y_j} * \Delta y_j \right)$$

The partial derivatives in the above equations can be computed from the projective transformation H or measured experimentally. The two linear equations can be represented in matrix notation as:

[2] $$\Delta x = T(\Delta y)$$

where the transform T is the 2 x J matrix of the partial derivatives of the xi with respect to the yj, over the J camera parameters.

To simplify our notation, we will assume that the image plane and 3-space coordinate axes have their origins at the projected and nominally imaged points respectively. Thus, the deltas in equation [2] can be dispensed with.

3. The Error Model

The multivariate normal probability density function has the form (for dimensionality "n"):

$$[3] \qquad P(X|U,M) = \frac{e^{(-.5 * (X-U)^T M^{-1} (X-U))}}{(2*\pi)^{\left(\frac{n}{2}\right)} * \sqrt{|M|}}$$

where: $U=E\{X\}$
$M=E\{(X-U)(X-U)^T\}$
$|A|$ = determinant of A.

The covariance matrix M must be positive semidefinite. That is, for any n-dimensional vector Z with real components we have:

$$[4] \qquad Z^T M Z \geq 0.$$

Theorem 1[1]:

If Y is distributed according to [3] with mean vector U and covariance matrix M, then:

If $X=TY+B$ with T a constant matrix and B a constant vector, then X is normally distributed with mean $V=TU+B$ and covariance matrix $W=E[(X-V)(X-V)^T]=TMT^T$

Thus, given our previously stated assumptions, we can now assert that the error distribution in the image plane will be a bivariate normal probability density function, having the same form as equation [3], but with mean vector V, and covariance matrix W, obtained as described in the above theorem.

---

[1] T.W. Anderson, An Introduction to Multivariate Analysis, p. 25, (John Wiley & Sons, New York, New York, 1958).

In more explicit form we have:

$$[5] \qquad P(x_1,x_2|0,0,\rho,s_1,s_2) = \frac{e^{\left(-\frac{G}{2}\right)}}{2 * \pi * s_1 * s_2 * \sqrt{1-\rho^2}}$$

where:

$$G = \frac{\left(\frac{x_1^2}{s_1^2} - \frac{2 * \rho * x_1 * x_2}{s_1 * s_2} + \frac{x_2^2}{s_2^2}\right)}{(1 - \rho^2)}$$

$$s_1 = \sqrt{E\{x_1^2\}} \qquad s_2 = \sqrt{E\{x_2^2\}}$$

$$\rho = E\left\{\frac{x_1 * x_2}{s_1 * s_2}\right\}$$

We note that ρ is the coefficient of correlation between x1 and x2 and $(-1 \leq \rho \leq 1)$.

The contours of constant probability density in the image {x1,x2} plane are the loci where the exponent of the density function is constant. They are similar coaxial ellipses, with their axes parallel to the eigenvectors of the covariance matrix W. In particular, the major axis of the ellipse will make an angle of

$$[6] \qquad \alpha = \frac{1}{2} * \text{ARCTAN}\left(\frac{2 * \rho * s_1 * s_2}{\left(s_1^2 - s_2^2\right)}\right)$$

with the x1 axis.

To simplify our derivation of the dimensions of the ellipse needed to provide a given level of probability of containing the image of the 3-space point being projected, we will transform our coordinate axes in the image plane so that they lie along the major and minor axes of the coaxial constant probability ellipses.

The resulting covariance matrix Q has the form:

$$[7] \qquad Q = \begin{pmatrix} q_1^2 & 0 \\ 0 & q_2^2 \end{pmatrix}$$

where the qi (the new variances) are the eigenvalues of the covariance matrix W. These eigenvalues are found by solving the following equation:

$$[8] \qquad 0 = \begin{vmatrix} (s_1^2 - q^2) & (\rho * s_1 * s_2) \\ (\rho * s_1 * s_2) & (s_2^2 - q^2) \end{vmatrix} .$$

The resulting solutions are:

$$q_1^2 = \frac{1}{2} * \left((s_1^2 + s_2^2) + \sqrt{(s_1^2 - s_2^2)^2 + 4 * \rho^2 * s_1^2 * s_2^2} \right)$$

[9] and

$$q_2^2 = \frac{1}{2} * \left((s_1^2 + s_2^2) - \sqrt{(s_1^2 - s_2^2)^2 + 4 * \rho^2 * s_1^2 * s_2^2} \right)$$

Substituting $q1^2$ for q^2 in either of the two homogeneous equations in:

$$[10] \qquad 0 = (W - q^2 * I) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

allows us to solve for the ratio of the x1 to x2 coefficient in the major eigenvector and determine its angle with the x1 axis to be:

$$[11] \qquad TAN(\alpha) = \frac{(q_1^2 - s_1^2)}{(\rho * s_1 * s_2)}$$

The above expression can be simplified using the identity ARCTAN(A)=2*ARCTAN({SQRT[1+A^2]-1}/A) to give the result in [6].

In terms of covariance matrix Q, the bivariate normal density function has the form:

$$[12] \qquad P(z_1,z_2) = \frac{e^{\left(-\frac{G}{2}\right)}}{2 * \pi * q_1 * q_2}$$

where: $$G = \frac{z_1^2}{q_1^2} + \frac{z_2^2}{q_2^2}.$$

The locus of $G=c^2$, where c is a constant is an equi-probability ellipse with major radius of length c*q1 and minor radius of length c*q2.

The area contained within this ellipse is c^2*q1*q2*PI and the differential area is 2*c*q1*q2*PI*Δc.

Thus, the probability p'' that the image of the nominally projected 3-space point will fall into the elliptic ring formed by the ellipses with parameters c and c+Δc is:

$$[13] \qquad P'' = c * e^{-\frac{c^2}{2}} * \Delta c.$$

Integrating p'' from 0 to c, we get:

$$[14] \qquad P = 1 - e^{-\frac{c^2}{2}}$$

where P is the probability that the image of the nominally projected 3-space point will fall into the ellipse with parameter c (i.e., the ellipse with major axis of length c*q1, minor radius of length c*q2, and orientation of the major axis of B; see equations [6] and [9] for the values of q1,q2, and α).

Some typical values for P are:

[15]

P	c
.50	1.177
.90	2.146
.95	2.448
.99	3.035

We note that if s1=s2=s, and =0, then q1=q2=s; the resulting contours are circles, and the parameter c corresponds to the radius of the resulting error circle measured in standard deviations (s). For this case, the radius which results in a 50% error probability is 1.177s, but the expected radial error is s*SQRT(PI/2)=1.253s, and the expected value of the square of the radial error is $E\{x1^2\}+E\{x2^2\} = 2*s^2$.

Finally, by invoking Bayes' theorem, we note that if an "error ellipse" as determined above is centered on the true projection of a given 3-space point, and has probability P of containing the actual projection of that point, then the same ellipse centered on the actual projection would have the same probability P of containing the true projection (assuming there is no difference in the way the true and actual projected points are distributed over the image plane).

Appendix B

RELATIVE UNCERTAINTY REGIONS

Appendix B

RELATIVE UNCERTAINTY REGIONS

Let p and q be two three-dimensional feature points. Let a1 represent an estimate of the camera parameters. Let F represent the perspective transformation, which is a function of the camera parameters, that maps feature points into image points. Then

[1] $P = F(a1,p)$ and $Q = F(a1,q)$,

where P and Q are the two-dimensional image coordinates of the points p and q. P and Q are the predicted image locations for the two features based on the estimates a1.

If an operator has correctly located the image of p at P', where should the image of q be? Or, in which region should the image of q appear? That is, what is the relative uncertainty region about q with respect to p and P'?

Assume that the actual camera parameters are a2 and the two features actually appear at P' and Q' in the image. Thus,

[2] $P' = F(a2,p)$ and $Q' = F(a2,q)$.

The relative uncertainty region can be described by the difference between (Q' - P') and (Q - P) as a function of a1 and a2.

Let

[3] $$a2 = a1 + \Delta a \ .$$

If we make the same assumption made in Appendix A that the parameter space is locally linear about a1 and a2, then

[4] $$P' = F(a1,p) + Mp * \Delta a$$

and

[5] $$Q' = F(a1,q) + Mq * \Delta a$$

where Mp and Mq are the 2 x N matrices of partial derivatives that describe the relative changes in the image plane as a function of the N camera parameters. Then

[6] $[(Q' - P') - (Q - P)] = Mq * \Delta a - Mp * \Delta a$

or

[7] $[(Q' - P') - (Q - P)] = (Mq - Mp) * \Delta a.$

If the Δa's are distributed according to a multivariate normal distribution, Theorem 1 in Appendix A applies. If the mean of the distribution is the vector U and the covariance matrix is S, the vectors on the left side of linear equation [7] will be distributed with mean V = (Mq-Mp)*U and covariance matrix W = (Mq-Mp)*S*(Mq-Mp)T.

Appendix C

AN ITERATIVE METHOD TO REFINE CAMERA PARAMETERS

Appendix C

AN ITERATIVE METHOD TO REFINE CAMERA PARAMETERS

The standard calibration problem is:

> Assume that the correspondence between world points and image points is a perspective transformation, G, that is a function of several camera parameters, such as the X, Y, and Z position of the camera, the heading, pitch, and roll of the camera, and the focal length of the camera. Given an initial estimate of the camera parameters and a set of world points (Xi,Yi,Zi) and their corresponding image locations (Ui,Vi), determine the best (according to some error metric) camera parameter values to map the world points into the image points.

If G is a linear function of the camera parameters and the square of the unresolved errors is used as the metric, there is a standard solution to the problem. Let G be represented as a matrix M. Then for each world and image point pair:

[1]
$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \end{pmatrix} \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix}.$$

A set of these equations can be combined into a single matrix:

[2]
$$\begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} X_1 & Y_1 & Z_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_2 & Y_2 & Z_2 \\ & & & \vdots & & \end{pmatrix} \begin{pmatrix} M_{11} \\ M_{12} \\ M_{13} \\ M_{21} \\ M_{22} \\ M_{23} \end{pmatrix}.$$

Let A be the vector of U's and V's, P be the matrix of X's, Y's, and Z's, and B be the vector of M's. Then [2] can be restated as:

[3] $$A = P * B$$

This equation can be directly solved for the best least-squares solution for D, whose elements are the six elements of the matrix M[1]:

[4] $$B = (P^T * P)^{-1} * P^T * A$$

Unfortunately, G is generally not linear. However, the least-squares solution of the linear problem can be embedded in an iterative solution to a nonlinear problem. The idea is to approximate the surface about the estimated parameter values by a hyperplane, solve that linear problem, and iterate until the desired precision has been achieved. If the hyperplane is determined by the partial derivatives of G with respect to the camera parameters, this approach is similar to a multidimensional Newton-Raphson method. See [Gennery][2] or [Bolles][3] for a more detailed description of this approach.

In our calibration method we consider G to be a function of the following camera parameters:

Cx, Cy, Cz---the position of the camera
Ch, Cp, Cr---the heading, pitch, and roll of the camera
Cf---the focal length of the camera
Su, Sv---the image scale factors for the U and V directions
Ir---the image rotation about the piercing point
Du, Dv---the U and V position of the piercing point

We group them into two categories: "internal" camera parameters and "external" camera parameters. The idea is that the internal camera parameters are functions of the camera itself and generally remain constant from one picture to the next. They are the image scale factors, the image rotation, the piercing point location, and the focal length. The external camera parameters specify the position and orientation of the camera and generally vary from one picture to the

---

[1] F.A. Graybill, An Introduction to Linear Statistical Models, Vol. I, (Mc Graw-Hill Book Company, 1961).

[2] Donald B. Gennery, "Least-Squares Stereo-Camera Calibration," Stanford Artificial Intelligence Project internal memo (1975).

[3] Robert C. Bolles, "Verification Vision within a Programmable Assembly System," Stanford University Ph.D. Dissertation (December 1976).

next. Since the focal length may change in a camera that has a zoom lens, it is sometimes treated specially. It is separated out of the list of internal camera parameters and treated like an external camera parameter to be computed.

We use homogeneous matrices to represent the transformations that are functions of the parameters listed above. The internal or "digitization" matrix is defined to be:

$$[5] \quad D = \begin{pmatrix} S_u & 0 & 0 & 0 \\ 0 & S_v & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(I_R) & \sin(I_R) & 0 & 0 \\ -\sin(I_R) & \cos(I_R) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -D_u \\ 0 & 1 & 0 & -D_v \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We assume that D is constant and given as a priori information.

G is defined as follows:

$$[6] \quad G = D*F*R*P*H*T$$

where

$$[7] \quad F = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{C_F} & 0 \end{pmatrix} \qquad T = \begin{pmatrix} 1 & 0 & 0 & -C_x \\ 0 & 1 & 0 & -C_y \\ 0 & 0 & 1 & -C_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$H = \begin{pmatrix} \cos(C_h) & \sin(C_h) & 0 & 0 \\ -\sin(C_h) & \cos(C_h) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(C_p) & \sin(C_p) & 0 \\ 0 & -\sin(C_p) & \cos(C_p) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} \cos(C_r) & 0 & \sin(C_r) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(C_r) & 0 & \cos(C_r) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

F is the perspective part of the camera transformation. T is the translation part. H, P, and R are the heading, pitch, and roll parts, respectively.

The transformation of world point (Xi,Yi,Zi) into an image point (Ui,Vi) is defined to be the following two-step computation:

$$[8] \qquad \begin{pmatrix} U_i' \\ V_i' \\ W_i' \\ S_i' \end{pmatrix} = D * F * R * P * H * \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}$$

$$U_i = \frac{U_i'}{S_i'} \qquad V_i = \frac{V_i'}{S_i'}.$$

In homogeneous coordinates S_i' is a scale factor for the vector and has to be divided out in order to obtain the image coordinates (Ui,Vi). Notice that Ui and Vi are not linear combinations of the camera parameters.

Given this representation of G, the partial derivative linear approximation to the surface in parameter space (about the initial estimates of the camera parameters) is:

$$[9] \qquad \begin{pmatrix} \Delta u_1 \\ \Delta v_1 \\ \Delta u_2 \\ \Delta v_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} \frac{\delta u_1}{\delta c_x} & \frac{\delta u_1}{\delta c_y} & \frac{\delta u_1}{\delta c_z} & \frac{\delta u_1}{\delta c_h} & \frac{\delta u_1}{\delta c_p} & \frac{\delta u_1}{\delta c_r} & \frac{\delta u_1}{\delta c_f} \\ \frac{\delta v_1}{\delta c_x} & \frac{\delta v_1}{\delta c_y} & \frac{\delta v_1}{\delta c_z} & \frac{\delta v_1}{\delta c_h} & \frac{\delta v_1}{\delta c_p} & \frac{\delta v_1}{\delta c_r} & \frac{\delta v_1}{\delta c_f} \\ \frac{\delta u_2}{\delta c_x} & \frac{\delta u_2}{\delta c_y} & \frac{\delta u_2}{\delta c_z} & \frac{\delta u_2}{\delta c_h} & \frac{\delta u_2}{\delta c_p} & \frac{\delta u_2}{\delta c_r} & \frac{\delta u_2}{\delta c_f} \\ \vdots & & & & & & \end{pmatrix} \begin{pmatrix} \Delta c_x \\ \Delta c_y \\ \Delta c_z \\ \Delta c_h \\ \Delta c_p \\ \Delta c_r \\ \Delta c_f \end{pmatrix}.$$

Since

$$[10] \qquad U_i = \frac{U'_i}{S'_i} \text{ and } V_i = \frac{V'_i}{S'_i} ,$$

the partial derivatives have the form:

$$[11] \qquad \frac{\delta u_i}{\delta C_x} = \frac{S'_i * \frac{\delta u'_i}{\delta C_x} - U'_i * \frac{\delta s'_i}{\delta C_x}}{S'_i * S'_i} ,$$

which depends on the partial derivatives

$$[12] \qquad \frac{\delta u'_i}{\delta C_x} \text{ and } \frac{\delta S'_i}{\delta C_x} .$$

These partial derivatives can be computed as follows:

$$[13] \qquad \frac{\delta u'_i}{\delta C_x} = \frac{\delta}{\delta C_x} D * F * R * P * H * T * \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}.$$

And since most of the matrices are constants with respect to the variables being differentiated, these expressions can be greatly simplified. For example:

$$[14] \qquad \frac{\delta u'_i}{\delta C_x} = D * F * R * P * H * \frac{\delta T}{\delta C_x} * \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}.$$

In summary, the iterative method to refine camera parameters is to compute the partial derivatives shown above, form the linear

approximation shown in [9] for the error surface, use the method discussed for equation [2] to solve this linear problem for corrections to be added to the current estimate of the camera parameters, use the corrections to form new estimates of the camera parameters, and iterate this process until the unresolved errors are sufficiently small.

For point-on-a-line matches, instead of two constraints per match (i.e., Ui error and Vi error), only one constraint is added to the list of constraints accumulated in the matrix shown in [9]. That one constraint is based on the perpendicular distance between the point in the image that is supposed to be on the line and the predicted image of the line.

The distance between a point in the image, (Ui,Vi) and a line that passes through the point (u0,v0) and at angle θ with respect to the U axis is:

[15] $$d = (U_i - U_0) \sin\theta - (V_i - V_0) \cos\theta .$$

Therefore, the constraint for a point-on-a-line match adds one line to the partial derivative linear approximation:

[16] $$\Delta d_i = \left(\frac{\delta d_i}{\delta C_x} \frac{\delta d_i}{\delta C_y} \frac{\delta d_i}{\delta C_z} \frac{\delta d_i}{\delta C_h} \frac{\delta d_i}{\delta C_p} \frac{\delta d_i}{\delta C_r} \frac{\delta d_i}{\delta C_f} \right) * \left(\Delta C_x \Delta C_y \Delta C_z \Delta C_h \Delta C_p \Delta C_r \Delta C_f \right)^T$$

where each entry has the form:

[17] $$\frac{\delta d_i}{\delta C_x} = \left(\frac{\delta u_i}{\delta C_x} \sin\theta - \frac{\delta v_i}{\delta C_x} \cos\theta \right) .$$

Each of these entries is a simple combination of the two partial derivatives used in the point-to-point case.

Notice that point-on-a-line matches and their constraints can be freely mixed with the normal point-to-point matches.

Appendix D

SYNTHESIS OF CLOUDS IN DATA BASE IMAGERY

Appendix D

SYNTHESIS OF CLOUDS IN DATA BASE IMAGERY

In order to test the Road Expert under adverse viewing conditions, we considered it necessary to acquire images containing various degrees of cloud cover. Our primary source of imagery, CALTRANS (California Department of Transportation), does not photograph roads during cloudy weather conditions and therefore we had to synthesize the clouds appearing in our road images.

In order to generate realistic clouds in our test imagery, the following criteria were established:

(1) Clouds should cast shadows.

(2) Edges of clouds should be controllably wispy--no hard edges. The same should be true of cloud shadows.

(3) Interior of clouds should be controllably transparent.

Prototypical clouds were extracted from digitized 70 mm U-2 photographs by subtracting from each pixel a constant level CTHRESH which removed virtually all of the background while leaving the clouds intact.

The cloud prototype image was:

$$CLOUD[i,j] = MAX\ [\ (U2image[i,j] - CTHRESH),\ 0\ ]$$

The following ramp function was introduced to satisfy b):

$$RAMP[i,j] = MIN\ [\ (CLOUD[i,j]/RAMPLEVEL),\ 1]$$

The ramp function assumes that cloud edges and partially transparent interiors of clouds have photometric levels close to zero (CTHRESH in the U-2 image). The "width" of the ramp is set indirectly by the selection of the intensity level "RAMPLEVEL."

Shadows are introduced by:

```
SHADOWIMAGE[i,j] = ROADIMAGE[i,j]
                  * (1 - (1 - GROUND.ATTEN) * RAMP[i+di,j+dj])
```

where di,dj define to offset of the cloud shadow with respect to the cloud. Clouds are assumed to be at a constant height above the underlying terrain. It is easily seen that when RAMP[i,j]=0, the image is unaffected, and when RAMP[i,j]=1, the image is attenuated by factor GROUND.ATTEN. Clouds are introduced to the shadow image by:

```
CLOUDIMAGE[i,j] = SHADOWIMAGE[i,j] * (1 - RAMP[i,j])
          + RAMP[i,j] * (CLOUD[i,j] * CLOUD.CONTRAST.FACTOR
                                    + CLOUD.INTENSITY.OFFSET)
```

This function smoothly blends the clouds with the shadowed road image according to the same ramp function.

The above procedure for synthesizing clouds has a total of seven parameters which control the attenuation of the ground intensity due to the cloud shadows and the clouds; control the blending at the cloud edges; control the relative contrasts of the clouds with respect to the ground; and finally, set the spatially offset of the cloud shadows with respect to the clouds.